entitled Database Frame Location below. As can be seen in the Table I,frames are stored in sequences which consist of an expression which proceeds from note Cm to note Cm#7, corresponding to a transition from event E1 to event E2. This consists of frames 1-420. Each of the 10 expressions in the database are stored.

In the next step, a key is assigned to each video frame corresponding to block 401 of Fig. 4. This is done by searching the database frame location database for the events E1 through E11. According to this step, a table such as Table III, entitled Record Database, set out below is computed. Thus, the first entry in the Table I corresponds to the first entry in Table III. The first event E1 is in original frames 0-60. The transition from E1 to E2 is in original frames 60-360, and the second event stored in frames 360-420. This sequence is divided so that new frame numbers 0-220 are stored with a key identifying them as transition E1 to E2.

In the next step, a content image as shown in Fig. 6 is generated based on the keys. This content image is generated by signing each key a position on the content image as shown on Table II below. Thus, event E1 will be assigned X,Y position (0,160) on the XY scale. Table II also shows the number of frames between each event on the display.

The final step, the video frames are compiled for addressing in response to the key corresponding to block 403 of Fig. 4. This step of compiling video frames results in an assembled database as shown below in Table IV. A given frame, e.g. frame 0 is compiled so it can be accessed in response to position (0,160) on the frame. In this embodiment, a sequence of frames is displayed in response to a user selecting the position which range from frame 0-220 as shown in Table II. In a more simple case as described above, only a single frame may be displayed in response to positioning of the cursor.

Once the content image of Fig. 6 is displayed and the assembled database of Table IV is computed, a user input devise may be used for accessing specific frames in response to a position on the content image as described with reference to block 404 of Fig. 4.

A1 These processing steps are thus executed by the processor 201 of Fig. 2 and interconnected by the databases and tables computed by the processing steps.

On page 25, line 14, before "Database Frame Location" please insert -- Table I. --

On page 26, please delete line 2 and insert therefore:

A2 The special coordinates of the content image are defined below in Table II.

Table II.

On page 26, line 19, before "Record" insert -- Table III. --

On page 26, line 35, before "Assembled" insert -- Table IV. --

On page 26, after the last line, please insert the following:

In Table IV, the first column corresponds to a given node, e.g. node E1 in the database. The second column identifies the number of nodes that are associated with the node in column 1. The third column identifies one of the adjacent nodes. The fourth column identifies the number of the frame associated with the first adjacent node identified in column 3. The fifth column identifies a second adjacent node, if any. The sixth column identifies the frame number associated with the second adjacent node. The seventh column identifies the X, Y coordinates on the content image of the node of column 1.

A3

32

a3

The eighth column identifies the audio which is to be produced in association with the node in column 1.